DOCUMENT RESUME

ED 427 653 IR 019 172

AUTHOR Brusilovsky, Peter; Schwarz, Elmar; Weber, Gerhard
TITLE A Tool for Developing Adaptive Electronic Textbooks on WWW.
PUB DATE 1996-10-00
NOTE 8p.; In: WebNet 96 Conference Proceedings (San Francisco, CA, October 15-19, 1996); see IR 019 168. Figure contains dark print.
AVAILABLE FROM Web site: http://aace.virginia.edu/aace/conf/webnet/html/151/151.htm
PUB TYPE Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Authoring Aids (Programming); Computer Software; Computer Uses in Education; Distance Education; *Electronic Text; *Hypermedia; Indexing; *Material Development; *Media Adaptation; Models; Navigation (Information Systems); *Textbooks; *World Wide Web
IDENTIFIERS *Authorware; Electronic Books

ABSTRACT

This paper describes an approach for developing adaptive electronic textbooks and presents InterBook, an authoring tool based on this approach which simplifies the development of adaptive electronic textbooks on the World Wide Web. The following topics are discussed: (1) content structuring, including the domain and user models, the glossary, and the annotated textbook; (2) functionality, including advanced navigation, adaptive navigation support, and prerequisite-based help; (3) authoring with InterBook, including a typical scenario for using this tool; and (4) indexing for more intelligent authoring on the Web. (Contains 15 references.) (DLS)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

Published on the Web from the WebNet-96 (forthcoming) proceedings with permission of the Association for the Advancement of Computing in Education (AACE).

ED 427 653

A Tool for Developing Adaptive Electronic Textbooks on WWW

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

Peter Brusilovsky
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
E-mail: plb@cs.cmu.edu

Elmar Schwarz and Gerhard Weber
Department of Psychology, University of Trier
D-54286, Trier, Germany
E-mail: {schwarz | weber }@cogpsy.uni-trier.de

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
G.H. Marks
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Abstract: An Electronic textbook is a popular kind of educational applications on World Wide Web (WWW). We claim that adaptivity is especially important for WWW-based educational applications which are expected to be used by very different groups of users without assistance of a human teacher . In this paper we describe an approach for developing adaptive electronic textbooks and present an authoring tool based on this approach which simplifies the development of adaptive electronic textbooks on WWW.

1 Introduction

World Wide Web opens new ways of learning for many people. Now, educational programs and learning materials installed and supported in one place can be used by thousands of students from all over the world. However, most of the existing educational WWW applications use the simplest solutions and are much more limited than existing 'on-site' educational systems and tools. For many designers, an ideal format of educational WWW material seems to be a static electronic copy of a regular textbook: chapter by chapter, page by page, picture by picture. Such electronic textbooks are non-adaptive, i.e., students with different abilities, knowledge, and background get the same educational material in the same form.

We claim that adaptivity is especially important for educational programs on WWW which are expected to be used by very different classes of users without assistance of a real teacher (who usually can provide adaptivity in a normal classroom). Currently, we can name very few adaptive educational applications on WWW [Brusilovsky, Schwarz & Weber 1996; Kay & Kummerfeld 1994; Lai, Chen & Yuan 1995; Nakabayashi et al. 1995]. All these applications keep a model of the user between sessions and use it to adapt the teaching sequence and the presentation of the material to a given user. The problem is that adaptive electronic textbooks are "knowledge-rich" applications and they are not very easy to design. There are some authorinig tools for developing "static" electronic textbooks on WWW [Goldberg, Salari & Swoboda 1996; Thimbleby 1996], but there are no tools available to support a designer in creating an adaptive textbook on WWW.

A possible approach for designing adaptive electronic textbooks on WWW was suggested recently in [Brusilovsky 1995]. This approach was further elaborated by the ELM research group in the University of Trier which applied it for developing an adaptive WWW-based LISP textbook ELM-ART [Brusilovsky, Schwarz & Weber 1996][Schwarz, Brusilovsky & Weber 1996]. Now we have generalized the experience gained in ELM-ART project and developed a subject-independent tool which simplifies the process of creating adaptive electronic textbooks on WWW. In this paper we describe our current approach for developing adaptive electronic textbooks on WWW and present an authoring tool which is based on this approach. In addition, we provide some recommendations for possible users of our tool and a brief review of relevant works.

IR019172

BEST COPY AVAILABLE

2 An Approach for Developing Adaptive Electronic Textbooks

Our approach to developing adaptive electronic textbooks on WWW based on the ideas from the areas of Intelligent Tutoring Systems [Wenger 1987] and Adaptive Hypermedia [Brusilovsky 1996]. Our adaptive textbooks use knowledge about its domain (represented in the form of domain model) and about its users (represented in the form of individual user models). The domain model serves as a basis for structuring the content of an adaptive ET. We distinguish two parts in an adaptive ET: a *glossary* and a *textbook*. Both these parts are based on the domain model. The student model is used by an adaptive ET to adapt its behavior to each particular user.

2.1 Content Structuring

The Domain Model and the User Model

According to our approach, the key to adaptivity in an adaptive ET are the domain model and the user model. The simplest form of domain model is just a set of domain concepts. By concepts we mean elementary pieces of knowledge for the given domain. Depending on the domain and the application area, concepts can represent bigger or smaller pieces of domain knowledge. A more advanced form of the domain model is a network with nodes corresponding to domain concepts and with links reflecting several kinds of relationships between concepts. This network represents the structure of the domain covered by a hypermedia system. The domain model provides a structure for representation of the user's knowledge of the subject. For each domain model concept, an individual user's knowledge model stores some value which is an estimation of the user knowledge level of this concept. This type of model (which is called an *overlay model*) is powerful and flexible: it can measure independently the user's knowledge of different topics.

The Glossary

The glossary is the central part of the ET. According to our approach, the glossary is considered as a visualized (and externalized) domain network. Each node of the domain network is represented by a node of the hyperspace, while the links between domain network nodes constitute main paths between hyperspace nodes. The structure of the glossary resembles the pedagogical structure of the domain knowledge and, vise versa, each glossary entry corresponds to one of the domain concepts. The links between domain model concepts constitute navigation paths between glossary entries. Thus, the structure of the manual resembles the pedagogic structure of the domain knowledge. In addition to providing a description of a concept, each glossary entry provides links to all book sections which introduce the concept. It means that the glossary integrates traditional features of an index and a glossary.

The Annotated Textbook

A human-written textbook represents human teaching expertise on how to introduce the domain concepts to the learners. It is usually a real textbook represented in hypermedia form. A textbook is hierarchically structured into units of different level: chapters, sections, and subsections. To make the textbook "more intelligent" and to connect it to the glossary, we have to let the system know what each unit of the textbook is about. It is done by indexing of textbook units with domain model concepts. For each unit, a list of concepts related with this unit is provided (we call this list *spectrum* of the unit). For each involved concept, the spectrum of the unit can represents also the role of the concept in the unit. Currently we support two roles: each concept can be either a *outcome* concept or a *prerequisite* concept. A concept is included in the spectrum as a outcome concept if some part of this page presents the piece of knowledge designated by the concept. A concept is included in the spectrum as a prerequisite concept if a user has to know this concept to understand the content of the page. Indexing is a relatively simple but powerful mechanism, because it provides the system with knowledge about the content of its pages:

the system knows which concepts are presented on each page and which concepts have to be learned before starting to learn each page. It opens the way for several adaptation techniques presented in the next subsection.

2.2 Functionality

Advanced Navigation

The knowledge about the domain and about the textbook content is used to serve a well-structured hyperspace. The system supports sequential and hierachical links between section. It generates the table of content where all entries are clickable links. In addition, it generates links between the glossary and the textbook. Links are provided from each textbook unit to corresponding glossary pages for each involved concept. On the other hand, from each glossary page describing a concept the system provides links to all textbook units which can be used to learn this concept. These links are not stored in an external format but generated on-the-fly by a special module which takes into account the user's current state of knowledge represented by the user's model. This approach is not only reducing page design time but also provides room for adaptation. In particular, our approach supports two adaptation techniques which have been applied in ELM-ART [Brusilovsky, Schwarz & Weber 1996]: adaptive navigation support and prerequisite-based help.

Adaptive Navigation Support

Our approach provides many more opportunities for browsing the course materials than traditional on-line textbooks. The negative side of it is that there is a higher risk for the user to get lost in this complex hyperspace. To support the user navigating through the course, the system uses *adaptive annotation*, an adaptive hypermedia [Brusilovsky 1996] technique. Adaptive annotation means that the system uses visual cues (icons, fonts, colors) to show the type and the educational state of each link. Using the user model, the system can distinguish several educational states for each page of material: the content of the page can be known to the user, ready to be learned, or not ready to be learned (the latter example means that some prerequisite knowledge is not yet learned). The icon and the font of each link presented to the user are computed dynamically from the individual user model. They always inform the user about the type and the educational state of the node behind the link. Red means not ready to be learned, green means ready and recommended, and white means no new information. A checkmark is added for already visited sections.

Prerequisite-based Help

The system knowledge about the course material comprises knowledge about what the prerequisite concepts are for any unit of the textbook. Often, when users have problems with understanding some explanation or example or solving a problem, the reason is that some prerequisite material is not understood well. In that case they can request prerequisite-based help (using a special button) and, as an answer to help request, the system generate a list of links to all sections which present some information about background concepts of the current section. This list is adaptively sorted according to the user's knowledge represented in the user model: more "helpful" sections are listed first. Here "helpful" means how informative the section is to learn about the background concepts. For example, the section which presents information about an unknown background concept is more informative then a section presenting information about a known concept. The section which presents information about two unknown background concepts is more informative then a section presenting information about one concept.

3 A Tool for Developing Adaptive Electronic Textbooks on WWW

3.1 Authoring with InterBook

Out tool (provisional name is InterBook) is aimed to help the author to transfer a normal textbook existing in electronic form into an adaptive ET. To get the most benefits from it an author should start creating an electronic textbook with a hierarchically structured MS Word file. This section demonstrates a typical scenario of using this tool for the case when the original textbook is available as MS Word file.
Step 1. Creating the list of domain concepts. Before starting to produce an adaptive ET, an author has to think about the list of domain concepts which will be used to annotate pages. An author does not have to have the list of *all* concepts before starting the work, it could be made when annotating the book.
Step 2. Structuring and Annotation. To let InterBook recognize the structure of a book, an author has to use the regular way of structuring an MS Word file. It means that the titles of the highest level sections should have a pre-defined text style "Header 1", the titles of its subsections should have a pre-defined text style "Header 2", and so forth. Then an MS Word file has to be annotated and indexed by the course author. An annotation is inserted into the file at the beginning of each section. The annotations have to be written using a special character style (hidden+shadowed). For each unit, the author can (but not have to) provide the set of outcome and background concepts. The format for the outcome annotation is: (out: concept-name1, concept-name2, ...).The format for the background annotation is: (pre: concept-name1, concept-name2, ...).
Step 3. Translation to HTML. The annotated MS Word file has to be saved in RTF formats and translated into an HTML file by the RTF2HTML program controlled by some specially designed settings. All annotations and section titles are translated into HTML comments which have a special format. The resulting "InterBook file" is just an HTML file annotated with several kinds of special comments.
Step 4. Parsing into LISP structure and Serving on WWW. When the InterBook server starts, it parses all InterBook files and builds the list of section frames. Each section frame contains the name and type of the unit, its spectrum, and its position in the original HTML file. The obtained LISP structure is used by InterBook to serve all the available textbook on WWW providing all advanced navigation and adaptation features. All content which the user will see on the screen is generated on-the-fly using the knowledge about the textbook, the user model, and HTML fragments extracted from the original HTML file. These features of InterBook are based on the functionality of he Common Lisp Hypermedia Server CL-HTTP.
As we can see, our tool seriously simplifies the design of adaptive ET on WWW for the authors who use the approach presented in [2 An Approach for Developing Adaptive Electronic Textbooks]. It provides full support in preparation and serving an ET for the authors who know only how to use the MS Word text processor. An advanced used who have some knowledge on HTML and LISP programming can use our tool more flexibly. For example, an author can bypass step 1 and 2 by preparing the textbook directly in HTML format with annotations provided as specially formatted comments. The author can also replace server response functions and HTML generating functions to implement different structure and different "look and feel" of the be requested by a unique URL. To enable the server to respond to a particular URL, this URL has to be associated to a response function implemented in LISP which has to generate an HTML page on the fly as an adaptive response. CL-HTTP includes a set of LISP functions for generating pages.

3.2 Working with InterBook

InterBook is expected to be used with Netscape 2.0 or 3.0 browsers. It uses advanced features of these browsers such as multiple windows and frames to provide the user with useful and powerful interface. Main windows used by InterBook are the textbook window and the glossary window.
The Glossary window is used to view the glossary. The upper part of the window is a list of glossary concepts. The lower part of this window is used to show the glossary entry for a concept. For each concept the system presents the concept description (if provided by the author), the list of section titles (selected from all available textbooks) which present the concept (i.e., which have it as an outcome concept) and the list of section titles which require this concept (i.e., which have it as a background concept). Section titles are clickable links which makes the corresponding section to be loaded to the Textbook window.
The Textbook window is the most important window in InterBook interface. This window is designed to view the main content of a textbook, section by section. It is divided into frames performing different functions. Main frame of the Textbook window is the Text window. This window shows a particular

section of the textbook which is called current section. For a terminal section the Text window shows the title of the section and the section itself. For a high-level section the Text window shows the title, the section preface (if existing) and the full table of content for the section (i.e. list of hierarchically structured titles of its subsections down to terminal level). A vertical bar to the right of the Text window is the Concept bar. It is used to show the concepts related with the current section. All names of concepts on the Concept bar are links to the Glossary. The upper part of the Textbook window hosts the navigation center and the toolbox. The navigation center shows the position of the current section in the textbook: it lists the titles of all direct predecessors (father, grandfather, etc.) and all brothers of the current section. All names of the sections are clickable links. The navigation center serves for both orientation and navigation. The toolbox provides a set of buttons which are used to call additional windows (such as content window, search window, and prerequisite-based help window) which provides additional functionality.

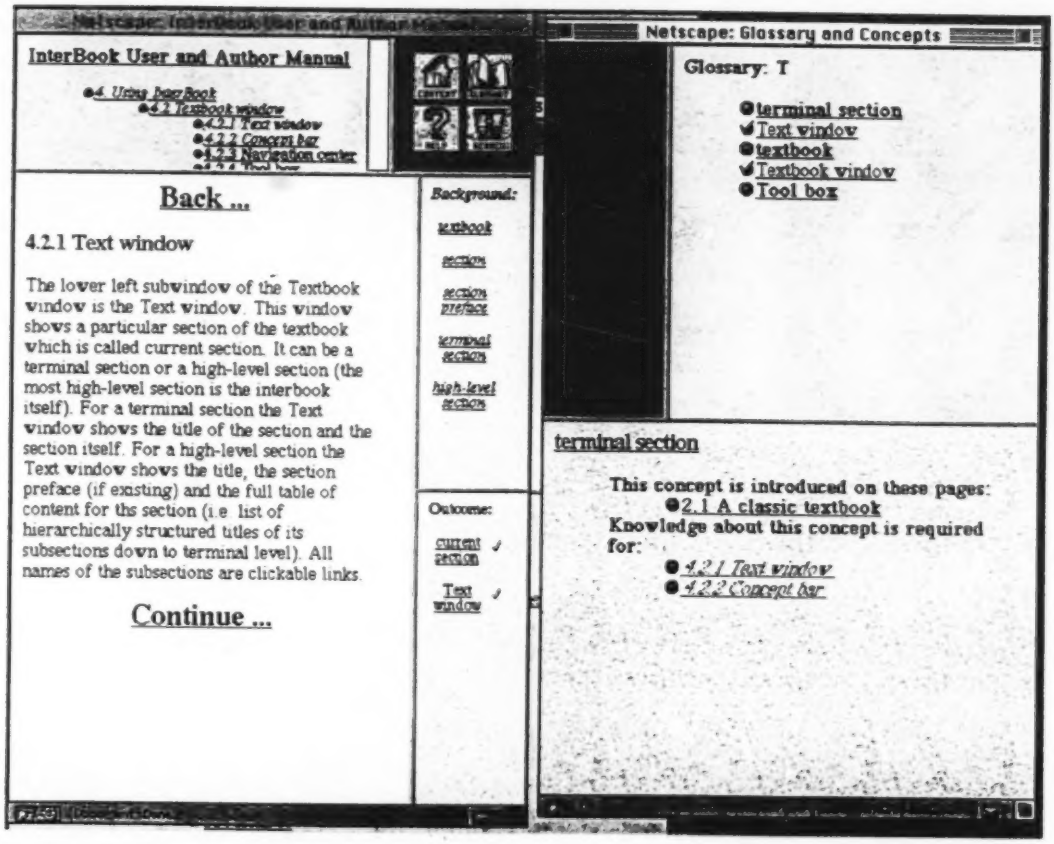
InterBook User and Author Manual
Back ...
4.2.1 Text window
The lover left subvindov of the Textbook vindov is the Text vindov. This vindov shovs a particular section of the textbook which is called current section. It can be a terminal section or a high-level section (the most high-level section is the interbook itself). For a terminal section the Text vindov shovs the title of the section and the section itself. For a high-level section the Text vindov shovs the title, the section preface (if existing) and the full table of content for ths section (i.e list of hierarchically structured titles of its subsections dovn to terminal level). All names of the subsections are clickable links.
Continue ...
Background:
Outcome:
Netscape: Glossary and Concepts
Glossary: T
terminal section
Text vindov
textbook
Textbook vindov
Tool box
terminal section
This concept is introduced on these pages:
2.1 A classic textbook
Knowledge about this concept is required for:
4.2.1 Text vindov
4.2.2 Concept bar

Figure 1: Textook and Glossary windows of InetrBook

4 Discussion: Indexing for "More Intelligent" Authoring on WWW

BEST COPY AVAILABLE

The main idea behind our tool is using concept-based *indexing* to make conventional educational material more intelligent and flexible. The idea of indexing is to provide the information about the content of each unit of conventional educational material by indexing this unit with related domain concepts. Previously, indexing was applied in three authoring contexts: CAI context, hypermedia authoring context and ITS authoring context.

Indexing was originally suggested in CAI context by Osin [1976] who suggested a framework for indexing CAI frames by a set of topics which it covers. Such indexed sets of frames were not related to any pre-scribed order of presentation. They can be accumulated, stored in special libraries, and re-used by different authors to create their own courses. In the multimedia field, a similar idea of a re-usable database of multimedia learning material indexed by topics and keywords is elaborated by Olimpo et al. [1990].

Later indexing was applied in hypermedia and ITS authoring area. In the hypermedia authoring area, an idea of indexing was elaborated by Mayes, Kibby and Watson [1988] in the StrathTutor system. They stressed additional preference of indexing the frames of learning material - the possibility to indicate related pairs of frames not by tedious glossary linking of pieces of learning material together, but dynamically, on the basis of similarity of corresponded sets of topics. In the ITS authoring area, indexing was applied to turn traditional CAI into a "slightly intelligent" ICAI [Elsom-Cook & O'Malley 1990; Grandbastien & Gavignet 1994; Vassileva 1992]. "Slightly intelligent" ICAI are based on both the CAI and ITS paradigms. The teaching material is not generated as in 'orthodox' ITS, but stored in CAI-like frames. However, these frames are indexed with the concepts from an explicit domain model network, so they can be selected intelligently. The most recent application of indexing on the crossroads of the above directions is hypermedia-based ITS which use indexing technology to connect the learning material represented in a hypermedia form with the domain knowledge base: SHIVA [Zeiliger 1993], ITEM/PG and ISIS-Tutor [Brusilovsky, Pesin & Zyryanov 1993].

Indexing shows to be a relatively cheap and useful technology for authoring "more intelligent" hypermedia and CAI systems. We argue that it is the relevant technology for developing more adaptive and intelligent educational applications of WWW. Currently, we can name only ELM-ART [Brusilovsky, Schwarz & Weber 1996] as an example of a WWW adaptive electronic textbook based on indexing. We expect that the WWW will boost the research and development work on adaptive electronic textbooks. We hope that our tool will be useful for those authors who are interested to make their ET adaptive and to serve it on WWW.

5 References

[Brusilovsky 1995] Brusilovsky, P. (1995). Adaptive learning with WWW: The Moscow State University Project. In Held, P., & Kugemann, W.F. (Ed.), Telematics for Education and Training . Amsterdam: IOS. 252-255.

[Brusilovsky 1996] Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. User Modeling and User-Adapted Interaction, 6 (2).

[Brusilovsky, Pesin & Zyryanov 1993] Brusilovsky, P., Pesin, L., & Zyryanov, M. (1993). Towards an adaptive hypermedia component for an intelligent learning environment. In Bass, L.J., Gornostaev, J., & Unger, C. (Ed.), Human-Computer Interaction (Lecture Notes in Computer Science, Vol. 753). Berlin: Springer-Verlag. 348-358.

[Brusilovsky, Schwarz & Weber 1996] Brusilovsky, P., Schwarz, E., & Weber, G. (1996). ELM-ART: An intelligent tutoring system on World Wide Web. In Frasson, C., Gauthier, G., & Lesgold, A. (Ed.), Intelligent Tutoring Systems (Lecture Notes in Computer Science, Vol. 1086). Berlin: Springer Verlag. 261-269.

[Elsom-Cook & O'Malley 1990] Elsom-Cook, M.T., & O'Malley, C. (1990). ECAL: Bridging the gap between CAL and intelligent tutoring systems. Computers and Education, 15 (1), 69-81.

[Goldberg, Salari & Swoboda 1996] Goldberg, M.W., Salari, S., & Swoboda, P. (1996). World Wide Web - course tool: An environment for building WWW-based courses. Fifth International World Wide Web Conference.

[Grandbastien & Gavignet 1994] Grandbastien, M., & Gavignet, E. (1994). ESCA: An environment to design and instantiate learning material. In de Jong, T., & Sarti, L. (Eds.), Design and production of multimedia and simulation-based learning material. Dordrecht: Kluwer Academic Publishers. 31-44.

[Kay & Kummerfeld 1994] Kay, J., & Kummerfeld, R.J. (1994). An Individualised Course for the C

Programming Language. Second International WWW Conference "Mosaic and the Web".
[Lai, Chen & Yuan 1995] Lai, M.-C., Chen, B.-H., & Yuan, S.-M. (1995). Toward a new educational environment. 4th International World Wide Web Conference.
[Mayes, Kibby & Watson 1988] Mayes, J.T., Kibby, M.R., & Watson, H. (1988). StrathTutor: The development and evaluation of a learning-by-browsing on the Macintosh. Computers and Education, 12 (1), 221-229.
[Nakabayashi et al. 1995] Nakabayashi, K. et al. (1995). An intelligent tutoring system on World-Wide Web: Towards an integrated learning environment on a distributed hypermedia. ED-MEDIA'95 - World conference on educational multimedia and hypermedia, AACE. 488-493.
[Olimpo et al. 1990] Olimpo, G. et al. (1990). On the concept of database of multimedia learning material. World Conference on Computers and Education, North Holland. 431-436.
[Osin 1976] Osin, L. (1976). SMITH: How to produce CAI course without programming. International Journal on the Man-Machine Studies, 8, 207-241.
[Schwarz, Brusilovsky & Weber 1996] Schwarz, E., Brusilovsky, P., & Weber, G. (1996). World-wide intelligent textbooks. ED-TELECOM'96 - World Conference on Educational Telecommunications, AACE. 302-307.
[Thimbleby 1996] Thimbleby, H. (1996). Systematic web authoring. Symposium "The Missing Link: Hypermedia Usability Research & The Web".
[Vassileva 1992] Vassileva, J. (1992). Dynamic CAL-courseware generation within an ITS-shell architecture. 4th International Conference, ICCAL'92, Springer-Verlag. 581-591.
[Wenger 1987] Wenger, E. (1987). Artificial intelligence and tutoring systems. Computational approaches to the communication of knowledge. Los Altos: Morgan Kaufmann.
[Zeiliger 1993] Zeiliger, R. (1993). Adaptive testing: contribution of the SHIVA model. In Leclercq, D., & Bruno, J. (Eds.), Item banking: Interactive testing and self-assessment. Berlin: Springer-Verlag. 54-65.

Acknowledgments

Part of this work is supported by a Grant from "Alexander von Humboldt Foundation" to the first author and by a Grant from "Stiftung Rheinland-Pfalz für Innovation" to the third author.